# E155 Course Syllabus

Welcome to E155! I am excited about all the cool projects you will make, and I am glad to have you here. This course teaches students how to practically realize complicated digital electronic systems, including FPGAs, microcontrollers and some chips of your own choosing. It culminates with a project where you decide what you want to build.

At the end of the course you will be able to

- Read technical documentation for microcontrollers and other digital chips
- Write and interpret clean RTL
- Design combinational logic
- Design finite state machines
- Identify and derisk clock domain crossings and other sources of asynchrony
- Program a microcontroller to interact with real-world systems (motors, speakers, etc.) using C
- Program a microcontroller using modern techniques like HAL or RTOS
- Decide when to use peripherals in a microcontroller circuit
- Name and describe common communication protocols and their tradeoffs
- Select parts for a large digital system, and decide how to spread computation among those parts
- Work with Github and present your work on a portfolio website
- Scope, design and implement a digital electronics project of your choice!

## Class Details

| Item | Information |
|---|---|
| Instructors: | Prof. Matthew Spencer |
| Lab Assistant(s): | Neil Chulani, Kavi Dey, Troy Kaufman, Vikram Krishna |
| Web page: | https://hmc-e155.github.io/ |
| Lab Checkoff Sheet: | Sheet Link |
| Email list: | We don't use email, only Discord |
| Discord Server: | See intro email for invite. |

Be sure to join the class Discord and check it regularly. It is be the only source of course-related communication for this semester.

| Name | Info |
|---|---|
| Lecture | TR 1:15 - 2:30 pm |
| Lab Checkoff | TWR afternoons by signup in the Digital Lab (PA B183) |
| Lab Hours | Saturday – Neil 2-4 PM, Kavi 6-8 PM<br>Sunday – Kavi 2-4 PM, Vikram 7-9 PM<br>Monday – Prof. Spencer 1-4 PM (check P2358 if not in lab), Troy 7-9 PM |

## Schedule

| Week # | Monday Date | Tuesday's Class | Thursday's Class | Due |
|---|---|---|---|---|
| 1 | 8/25 | Intro & Analog Behavior of Digital Systems | Combinational and Sequential Logic | Git & Quarto Portfolio Setup |
| 2 | 9/1 | Verilog Coding | Synchronous Design | Lab 1 |
| 3 | 9/8 | FPGA Documentation | C Programming on an MCU | Lab 2 |
| 4 | 9/15 | GPIO | Clock Configuration | Lab 3 |
| 5 | 9/22 | Timers | Interrupts | |
| 6 | 9/29 | PCB Design Activity | PCB Design Activity | Lab 4 |
| 7 | 10/6 | Serial Interfaces Overview, SPI, and Project Kickoff | UART and the IoT | Lab 5 |
| 8 | 10/13 | **Happy Fall Break! No Class** | Proposal Debriefs | Project Proposal |
| 9 | 10/20 | Advanced Encryption Standard (AES) | AES on FPGA Workshop | Lab 6 |
| 10 | 10/27 | Graphics and Displays | Motors and Speakers | Lab 7 |

| Week # | Monday Date | Tuesday's Class | Thursday's Class | Due |
|---|---|---|---|---|
| 11 | 11/3 | Design Review Presentations | Design Review Presentations | Design Review Presentation & Memo |
| 12 | 11/10 | Introduction to Real Time Operating Systems | Direct Memory Access | |
| 13 | 11/17 | Project Status Report and Demo | Connectors | Project Status Reports & Demo |
| 14 | 11/24 | The Fast Fourier Transform (FFT) | Happy Thanksgiving! **No class** | |
| 15 | 12/4 | Digital Signal Processing | Special Topics / Guest Lecture | Project Checkoffs, Report, Demo Day |

You will be working on labs on your own time and it is not required that you attend the entire scheduled lab period. Instead, sign up for a time to get your lab checked off. Please sign up for a time during your lab section. If you are unable to find a spot that works for you, see if you can swap with one of your classmates. If you are still having trouble finding a time that works for you, reach out and let me know.

## Recommended Texts

I assume you have a solid grasp of digital design at the level of E85; review Digital Design and Computer Architecture, ARM Edition (link) if you feel rusty on a topic. We will not be closely following a textbook, but you may find The Definitive Guide to ARM® Cortex®-M3 and Cortex®-M4 Processors by Joseph Yiu to be a helpful, comprehensive reference for the microcontroller. An electronic version of the book is available through the library (link).

## Lab Kit

While there is not a textbook to purchase, you will need to buy a lab kit. The fee is $75 in Claremont Cash, and should be paid by filling out the Google Form (link) which authorizes Sydney Torrey in the Engineering office to charge your Claremont Cash account. Once you have paid for your kit via the form, see Jacob Staimpel in the stockroom to pick up your kit.

If you cook your board this semester, you can buy and rebuild a replacement, but ask the instructor for help troubleshooting first. You'll also check out a large breadboard from the stockroom, and will need to return it at the end of the semester.

The kit fee can be waived in cases of financial hardship. To request a waiver fill out the form here (link). Course instructors will not know about waiver requests.

You will also need to check out a breadboard for the semester. You can do so by talking to Jacob Staimpel in the stockroom.

## Lab Access

The Digital Lab (Parson B183) is available for you to use when working on your labs. The current door code will be shared privately. There are Windows PCs available with SEGGER Embedded Studio and Lattice Radiant installed along with the drivers required to program your board. The lab also has the electronics assembly equipment needed to solder, oscilloscopes and power supplies at the lab stations, and a lab cabinet with various resistors and some of the parts like wires, seven-segment LEDs, and transistors you will need for some of your labs. You are welcome to use these while working on your lab, but please make sure to return the components to the lab cabinet when you are done.

In addition, the software we will be using for programming the MCU (SEGGER Embedded Studio for ARM) and FPGA (Lattice Radiant) are free and supported on a variety of platforms if you wish to download them on your personal computer. SEGGER Embedded Studio is supported on Windows, MacOS, and Linux and Lattice Radiant is supported on Windows and Linux. If you are running MacOS, you can download and virtualize Windows using VMWare Fusion Pro under a Personal Use License for free. More details and download links can be found here (link).

## Grading

The grading for this class is based on a variation of specifications-grading and may be a bit different than what you have seen in other classes. The grade you earn in the class will be determined based the number of deliverables you successfully complete and the level of polish to which you complete them across three different bundles:

1. Labs
2. Project
3. AI reflections
4. Completion-graded assignments: the pre-assessment, the post-assessment and setting up your portfolio

Each assignment will contain a list of specifications (or specs) for two levels of completeness: proficiency and excellence. The list of specifications are designed to be aligned with the learning goals for the assignment. The proficiency specifications will indicate the level of completeness that demonstrates that you have achieved a level of comfort with the material in the assignment such that you would be able to implement the learning outcomes in a different setting. Excellence specs are above and beyond proficiency specs. Meeting the excellence specs for an assignment indicates that you have not only achieved the basic level of expected knowledge of the material, but have truly understood and are able to apply the techniques with deftness.

The table below describes the levels that you need to meet in each bundle (i.e., column) in order to earn the grade in the respective row. To earn the grade in a given row, you must meet **all** the criteria in that row. In other words, to get a B, you must complete the following:

- 6 labs to the proficiency specs and 5 of those labs must meet the excellence specs.
- All 5 project elements must meet proficiency specs and 4 of them must meet the excellence specs.

| Grade | Labs | Project |
|---|---|---|
| F | $P < 4$ | $P < 4$ |
| D | $P \geq 4$, $E \geq 0$ | $P \geq 4$ |
| C | $P \geq 5$, $E \geq 1$ | $P \geq 4$, $E \geq 2$ |
| B | $P \geq 6$, $E \geq 4$ | $P = 5$, $E \geq 4$ |
| A | $P \geq 6$, $E \geq 6$ | $P = 5$, $E = 5$ |

Performance on the labs and project is weighted equally. +/- grades will be assigned for situations in which your performance falls between the conditions for each row.

As one example, if you meet the requirements for an A in the labs bundle but the requirements for a B in the project bundle, you can expect to earn either a B+ or A-. In these situations, your in-class participation, AI reflections, and completion of completion-graded assignments will help decide whether you split the difference between the - of the higher tier and the + of the lower tier.

An essential part of specifications grading is allowing you to retry specifications if they fail, and allowing flexibility is an important part of helping classes to effectively serve students. To that end, each student has one late token and two redo token that allow them to respectively, turn in a lab up to a week late (checkoff in office hours) and redo a lab where they did not achieve a specification they desired.

## Collaboration Policy

Your peers are an excellent source of support and can be a great help as you complete MicroPs. With that said, it is important that each student in the class do the work for themselves to develop their own expertise. The collaboration policy for MicroPs is as follows:

> **i** Collaboration Policy
>
> You **should**:
>
> - Discuss high level questions about the approach to the lab **after** you have spent time developing your own initial approach.
> - Ask for feedback on your Verilog code **after** both you and the person you are asking to review your code have made a significant attempt on your own.
> - List on your lab writeup any people that have reviewed your work or for whom you have reviewed work.
> - Ask the instructor any question you think might be inappropriate to discuss with a peer.
>
> You may **not**:
>
> - Pair program.
> - Use code that you have not personally typed (e.g., do not copy-paste code from any source). Note that this policy does not apply to the portions of your lab where you are encouraged to use AI.
> - Directly copy code from any other person.
> - Use any source code in your project without citation (e.g., if you use a module from a textbook or another online source, be sure to cite it in the comments of your code.).

## AI Policy

Harvey Mudd is piloting AI/LLM use language that instructors can adopt in their syllabi. During the labs in this course, we are closest to level 0 - no AI use allowed. During the project we are closest to level 3 - full AI use allowed.

Academic work on the effects of AI on learning are still nascent, but a case study in a course similar to ours suggested that students who rely on AI early in the course often struggle with concepts later in the course. However, LLMs are rapidly transforming industrial practice in software development and digital engineering, and other academic work suggests that an AI with expert prompting can succeed in programming a microcontroller to the level required by a class.

Therefore, this course is taking a cautious stance about engagement with AI. Many of the learning goals (especially learning to interpret documentation and RTL) are undermined by the use of AI. Therefore, in the policy outlined below, AI will be allowed only on "AI prototype" questions in the labs. During the project, I am amenable to more liberal AI use because you will have demonstrated many fundamental skills from the class. We can discuss more at the project launch.

**Using AI in MicroPs**

In MicroPs this fall we will adopt the following policies for generative AI use during labs 1-7:
You **should**:

- **Use AI on the AI prototypes throughout the course.** Each lab this fall will include a section describing a generative AI experiment. For the tasks in these sections you are required to leverage generative AI. As you embrace AI, I would encourage you to reflect on your experience using it and on its impact on your skill development. Your reflections on these AI experiments each week are a great topic to discuss in your weekly reflections.

You should **not**:

- **Use generative AI to directly generate any code on the main lab design challenges.** Learning how to write Verilog and C code is a core learning objective in this class. The best way to learn how to evaluate the quality of LLM-generated code is by first writing a large quantity of it for yourself.
- **Use generative AI to generate any of your technical documentation.** Thinking is writing, and learning how to articulate what you've done and what it means matters. Outsourcing this task to an LLM will not help you to better understand the work you are doing. As noted above, you may use spellcheckers, but you must not use tools which will rewrite sentences for you or generate text (e.g., Google's "Help Me Write" feature)
- **Use generative AI to summarize or search datasheets** Learning to read datasheets is a foundational and essential skill in this discipline, and verifying AI's work is much harder if you haven't read datasheets on your own at first.

## Honor Code Policy

Students in this class are expected to follow the HMC honor code. An honor code policy appears below and prescribes behavior that is considered honorable, so read those maxims and follow them closely. Any honor code violations will be handled through JB.

If you believe you have violated the honor code in any way, please take the initiative to self report so that we can come up with a fair and just path forward together.

1. All students enrolled in this course are bound by the HMC Honor Code. More information on the HMC Honor Code can be found in the HMC Student Handbook.
2. It is your responsibility to determine whether your actions adhere to the HMC Honor Code. If this document does not clarify the legitimacy of a particular action, you should contact the course instructor and request clarification.
3. Work you submit for individual assignments should be your own, and you should complete all assignments based on your own understanding of the underlying material. If you work with, or receive help from, another individual on an assignment, provide a written acknowledgement in complete sentences that includes the person's name and the nature of the help.
4. This document is not meant to be an exhaustive list of every possible Honor Code violation. Infractions not explicitly mentioned here may still violate the Honor Code.
5. **Boundaries of Collaboration.** Verbal collaboration with other students on individual assignments is encouraged **after** you have given serious thought to each component yourself. However, all submitted written work should be written by yourself individually, and not a collaborative effort or copied from a common source (e.g., a chalkboard). It is **not** acceptable to work on labs in lockstep with another classmate.
6. **Use of Computer Software.** The use of graphing calculators and computer software to aid in course work is acceptable, as long as it does not substitute for an understanding of the course material.
7. **Use of Web Resources.** The use of Internet resources to aid in course work is acceptable, as long it does not substitute for an understanding of the course material. Plagiarism and direct copying from online (or any other) sources is strictly prohibited.
8. **Use of Your Own Work from Previous Semesters.** If you have previously attempted this course, you may resubmit your work from previous semesters as this semester's coursework, as long as you understand the underlying material.
9. **Use of Other Course Resources from Previous Semesters.** You may not reference assignments (labs, problem sets, activities) of this course from previous semesters.
10. **Retention of Course Resources.** Assignments and exams from this course may not be committed to dorm repositories or otherwise used to help future students.

## Inclusiveness and Harassment

We do difficult work in this class and everyone should feel comfortable engaging with the material. We explicitly want you to feel safe doing this work, so it is worth stating that the instructors are committed to making the class a safe space for everyone regardless of race, gender, ethnicity, sexual orientation, religion, and academic history. If you feel that you are experiencing a hostile environment, speak to an instructor immediately.

## Educational Accessibility

HMC is committed to providing an inclusive learning environment and support for all students. Students with a disability (including mental health, chronic or temporary medical conditions) who may need some accommodation in order to fully participate in this class are encouraged to contact Educational Accessibility Services at ability@g.hmc.edu to request accommodations. Students from the other Claremont Colleges should contact their home college's disability resources officer.